Table of Contents

Attorney's Docket No. 10005473-1
(PATENT)

AF

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

In re Patent Application of	
Teddy Christian Johnson	Group Art Unit: 2153
Application No.: 09/862,538	Examiner: Krisna Lim
Filed: May 21, 2001	Appeal No.:
For: METHODS AND STRUCTURE FOR IMPLEMENTING WEB SERVER QUALITY OF SERVICE CONTROL	

SUPPLEMENTAL APPEAL BRIEF

Mail Stop APPEAL BRIEF - PATENTS
Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

Sir:

This Supplemental Appeal Brief is in response to a Notification of Non-Complaint Appeal Brief dated December 5, 2006. The Notification asserts that the previously filed Appeal Brief fails to comply with the previous Appeal Brief set forth under 37 C.F.R. §41.37(c). In particular, this Supplemental Appeal Brief includes revised Summary of Claimed Subject Matter and Grounds of Rejection to be Reviewed on Appeal sections. Because the appropriate Appeal fees were paid on March 1, 2006, no fees are due at this time.

I. Real Party in Interest

The real party in interest for this appeal is: Hewlett-Packard Development Company, a Texas Limited Liability Partnership having its principal place of business in Houston, Texas.

II. Related Appeals and Interferences

There are no other appeals, interferences, or judicial proceedings which will directly affect or be directly affected by or have a bearing on the Board's decision in this appeal.

III. Status of Claims

A. Total Number of Claims in Application

There are 17 claims pending in application.

B. Current Status of Claims

1. Claims canceled: None
2. Claims withdrawn from consideration but not canceled: None
3. Claims pending: 1-17
4. Claims allowed: None
5. Claims rejected: 1-17

IV. Status of Amendments

There are no pending amendments to the claims.

V. Summary Claimed Subject Matter

The method for providing Web services as recited in claim 1 comprises a number of steps. With reference to Figure 2 and the related disclosure at pages 7-8 of the Specification, the steps include a computing node (Server 120, page 6, line 9) receiving a request from a Web client process wherein said request includes

customer ID information (Figure 2 200, page 7, lines 11-12) and spawning (Figure 2 206, page 8 lines 6-10) a program element operable on the computing mode to process the request. The method also includes spawning a program element operable on the computing node to process the request and associating the customer ID information with the spawned program. (See page 4, lines 4-6 and 20-22, page 8, lines 16-22). An additional step includes allocating computing resources of said computing node to the spawned program element in accordance with said customer ID information associated with the request (page 9 line 6 to page 10, line 7).

The system for delivery of services in a client/server distributed environment as recited in claim 8 is generally described in Figure 1 and the related disclosure at page 6, line 9 to page 7, line 6 of the Specification. In the system, a server 110 is operable as a process on a server computing node 100 to process requests from a plurality of client processes (102s in Figure 1) coupled to the server computing node (120 in Figure 1). The server process (See 120 Session Monitor). The system also includes a plurality of server child processes operable on said server computing node (See 112 Log of Sessions) in which each child process is associated with a customer ID and spawned by the server process to process the requests from said plurality of client processes (See page 8, lines 16-17; Page 5, lines 2-4, Page 8, lines 25-26). A process resource manager (See 120 Session Monitor) on the server computing node controls allocation of resources of the server computing node among the plurality of server child processes in accordance with the customer ID associated with each process (See page 5, line 12 to line 16, page 6, line 25 to page 7, line 6, and page 9, line 21 to page 10, line 7.)

In one embodiment of the system as recited in claim 9, the server process is a Web server process, and each child process is a cgi-bin process. (See page 6, lines 16-20).

In another aspect of the system as recited in claim 10, the processor resource manager allocates CPU time, secondary storage bandwidth, and main memory to each child process. (Page 6, lines 29 to page 7, line 6.)

Independent claim 11 is written in means plus function format. The support in the specification for each of the functions of the elements of claim 1 also provide

examples of support in the specification for the functions described in the means plus function elements of claim 11. A structure corresponding to the means for receiving a request from a Web client process wherein said request includes customer D information is the server computing node of Figure 1 described on pages 6-7 with reference to Figures 1 and 2. An example of corresponding structure for spawning a program element operable on a computing node to process said request (See page 6, liens 16-17), associating said customer ID information with the spawned program (See page 4, lines 4-6 and 20-22, page 8, lines 16-22), and for allocating computing resources of said computing node to the spawned program element in accordance with said customer ID information associated with said request is the resource monitor element (Session Monitor in Figure 2) described above (See page 5, line 12 to line l6, page 6, line 25 to page 7, line 6, and page 9, line 21 to page 10, line 7.)

Regarding claims 12 and 13, the resource monitor (Session Monitor in Figure 1) as discussed on page 9, line 21 to page 10, line 7 would be an example of corresponding structure for the means for allocating a minimum level of resources to the spawned program element in accordance with said customer ID information and the means for allocating a maximum level of resources to the spawned program element in accordance with said customer ID information.

With respect to claims 15-17, the resource monitor (Session Monitor in Figure 1) as discussed on page 6, line 29 to page 7, line 6 would be an example of corresponding structure for the means for allocating processor time utilization, means for allocating main memory utilization and the means for allocating secondary storage bandwidth utilization.

VI. Grounds of Rejection to be Reviewed on Appeal

The Claims on Appeal were rejected in the final Office Action under the following grounds:

A. Claims 1-6 and 11-16 are rejected under 35 U.S.C. § 102(e) as anticipated by *Zenner* (U.S. Patent No. 6,718,330).

B. Claims 7-10 and 17 are rejected under 35 U.S.C. § 103(a) as unpatentable over the *Zenner* patent in view of *Cherkasova et al* (U.S. Patent No. 6,865,601).

VII. Argument

A. The *Zenner* Patent Fails To Teach All Of The Recited Claim Elements

To anticipate a claim under 35 U.S.C. § 102, a reference must teach every element of the claim. See *Verdegaal Bros. v. Union Oil Co. of California*, 814 F.2d 628 (Fed. Cir. 1987). Moreover, in order for an applied reference to be anticipatory under 35 U.S.C. § 102 with respect to a claim, "[t]he identical invention must be shown in as complete detail as is contained in the . . .claim." See *Richardson v. Suzuki Motor Co.*, 9 U.S.P.Q.2d 1913 (Fed. Cir. 1989).

Independent Claim 1

Claim 1 recites the following:

> A method for providing Web services comprising the steps of:
> a computing node receiving a request from a Web client process wherein said request includes customer ID information;
> spawning a program element operable on the computing node to process said request;
> associating said customer ID information with the spawned program; and
> allocating computing resources of said computing node to the spawned program element in accordance with said customer ID information associated with said request.

As illustrated in its Figure 3, *Zenner* discloses a prediction process (Col. 6, lines 7-25) for initially assigning a user request to an agent workstation based on the expected processing time of the request by that agent and provides for reassignment of the request based on predetermined criteria such as the user request has not been processed within a certain time period. (See Col. 8 lines 34-4 1). A user request comes in a form such as an e-mail, a form, or a recorded voice message. (Col. 6, lines 9-1 1). The basis of the initial assignment are based on a number of factors (e.g., experience level of the agent, expertise level of skill set of the agent, etc. see Col. 6, lines 40-64) related to the agent's expected time for processing the request. "To predict which agent is most likely to finish servicing the request first, the Pre-IAWD 220 determines and uses characteristics of the request such as a priority of the request, a subject matter or topic of the request, and an average time the group of agents takes to service the request." (Col. 6, lines 27-31).

In the Final office action, the Examiner cites for support in his paragraph 15 on page 6 the following:

b) spawning (distributing) a program element (agent workstations of Fig. 2, number of available agents at a given time (col. 6) operable on a computing node to process said request (e.g., see Fig. 2, col. 5 (line 51) to col. 6 (line 64)).

Appellant respectfully reasserts the previously presented argument in his response of August 16, 2005 that the term "spawn" has a particular meaning in the computer science context; namely to create a child program element by another program element, i.e., the parent program element. A typical embodiment of a program element is a process, but not a workstation. For a 35 U.S.C. 102(e) rejection, the reference has to disclose each and every element of the claim. The description for predicting which agent, a human being or robotic response (*Zenner*, Col. 5, lines 63-64) is going to provide the quickest customer service based on characteristics such as the topic of the request and the experience and skill level of the agent is solving a different problem at a level of abstraction not concerned with, and moreover, not disclosing spawning a program element. *Zenner* does not disclose each and every element of claim 1; hence claim us patentable over *Zenner*.

Independent Claim 11

The arguments presented for claim 11 are also applicable for establishing that *Zenner* does not disclose a "means for spawning a program element operable on a computing node to process said request." *Zenner* fails to disclose the invention as recited in claim 11 as it is contained in the claim. Thus, *Zenner* does not anticipate claim 11, and Appellant respectfully requests that the Board overturn this rejection.

Claims 2-6 and 12-16

Claims 2-6 depend from claim 1, and claims 12-16 depend from claim 11. Thus each of claims 2-6 and 12-16 inherit all of the limitations from its respective base claim, and each of claims 2-6 and 12-16 contain limitations not taught by *Zenner*. Therefore, Appellant respectfully asserts that *Zenner* does not anticipate claims 2-6 and 12-16 either and respectfully asks this Board to overturn the rejections.

Furthermore, *Zenner* fails to disclose "allocating a minimum level of resources to the spawned program element in accordance with said customer ID information" of claim 2 or "allocating a maximum level of resources to the spawned program element in accordance with said customer ID information." The resources in these claims are referencing the computing resources of claim 1. *Zenner* provides a prediction scheme for assigning a request to a particular agent based on a variety of real world factors as described in Col. 6, lines 40-64 including the user identification, expertise and experience levels, average time for processing a request etc. However, *Zenner* is not providing a solution at the level of computing resources. It only discloses how to assign a request to an agent but not a level of computing resources for a spawned program element based on the customer ID information. The arguments with respect to claims 2 and 3 are also applicable to claims 12 and 13. In view of these additional reasons as well, the Board is respectfully requested to overturn the rejection of claims 2-3 and 12-13.

B. The *Zenner* And *Cherkasova* Patents When Applied Individually Or In Combination Fail To Teach Or Suggest All Of The Recited Claim Elements

To establish a prima facie case of obviousness, three basic criteria must be met. First, there must be some suggestion or motivation, either in the references themselves or in the knowledge generally available to one of ordinary skill in the art, to modify the reference or to combine reference teaching. Second, there must be a reasonable expectation of success. Finally, the prior art reference (or references when combined) must teach or suggest all the claim limitations. The teaching or suggestion to make the claimed combination and reasonable expectation of success must be found in the prior art, and not based on applicant's disclosure. MPEP §2143, Rev. 7 (August 2006).

Claims 7, 8-10, and 17

First, one of ordinary skill in the art would not be motivated to combine *Zenner* and *Cherkasova* to make the claimed invention. In the Final Office Action, the Examiner states:

> Moreover, both of these two references are directed to the distributing of workload in the computer network. Thus, it would have been obvious to one of ordinary skill in the art to combine the teaching of *Cherkasova* into *Zenner* in order to receive the Internet workload distributor that can distribute equally workload among the servers.

The two references are not directed to the distributing of workload in the computer network, nor to distributing equally workload among servers. *Cherkasova* discloses load balancing among server nodes in a server cluster. More specifically, *Cherkasova* discloses organizing web sites into groups and then assigning each group of web sites to one of the server nodes in the server cluster. *Zenner* is directed to distributing customer requests in the form of e-mails or recorded voice messages to the agents typically human beings based on non-computer implemented criteria such as the experience and skill sets of the agents and the nature of the real-world problem the customer is having as determined from the customer's request. The problems being solved are different; hence, the disclosures of *Cherkasova* and *Zenner* are describing different, quite unrelated solutions. Hence, one of ordinary

skill in the art is not going to be motivated to combine these references to make the invention as claimed in claims 8-10, or claims 7 and 17 for that matter.

Claims 8-10

Furthermore, the resulting combination does not disclose each and every element of claims 7, 8-10, and 17. Accordingly, *Cherkasova* fails to overcome the deficiencies of *Zenner* described above with respect to Claims 1-6 and 11-16. For example, *Cherkasova* fails to disclose or suggest details regarding operations within a server node, and particularly fails to disclose or suggest

> a plurality of server child processes operable on said server computing node and spawned by said server process to process said requests from said plurality of client processes wherein each child process of said plurality of server child processes is associated with customer ID information.

Thus, independent claim 8 is patentable over *Zenner* in view of *Cherkasova*. Claims 9-10 depend from claim 8; thus, each of claims 9-10 inherit all of the limitations from its respective base claim. Therefore, Appellant respectfully asserts that claims 8-10 are patentable over *Zenner* in view of *Cherkasova* and respectfully asks this Board to overturn the rejections.

Claims 7 and 17

Claim 7 depends from claim 1 and claim 17 depends from claim 11, and each inherits all of the limitations from its respective base claim. Thus, each of claims 7 and 17 are not anticipated by *Zenner*. As noted in paragraph 12, page 4 of the Final Office Action, neither of *Zenner* or *Cherkasova* disclose that the computing resources include secondary storage bandwidth utilization. A link between a LAN 122 and the Internet 128 in Figure 1 of *Zenner* may suggest that there is in general bandwidth between networks, but would not suggest the additional feature of the claims as detailed in the claim language, namely that computing resources allocated in accordance with the customer ID information include secondary storage bandwidth utilization.

In summary there is no suggestion within the references themselves to combine them to make the claimed invention. Moreover, the combination does not disclose each and every element of claims 8-10 and 7 and 17. Therefore, Appellant

respectfully asserts that claims 7-10 and 17 are patentable over *Zenner* in view of *Cherkasova* and respectfully asks this Board to overturn the rejections.

VIII. Claims Appendix

See attached Claims Appendix for a copy of the claims involved in the appeal.

IX. Evidence Appendix

No evidence pursuant to §§ 1.130, 1.131, or 1.132 or entered by or relied upon by the examiner is being submitted.

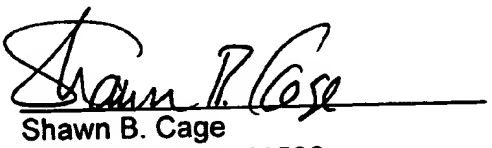
X. Related Proceedings Appendix

No related proceedings are referenced in II. above, or copies of decisions in related proceedings are not provided, hence no Appendix for this section is included.

Respectfully submitted,

BUCHANAN INGERSOLL & ROONEY PC

Date January 5, 2007

By: 
Shawn B. Cage
Registration No. 51522

P.O. Box 1404
Alexandria, VA 22313-1404
703 836 6620

VIII. CLAIMS APPENDIX

The Appealed Claims

Claims Involved in the Appeal of Application Serial No. 09/862,538

1. (Previously Presented) A method for providing Web services comprising the steps of:

a computing node receiving a request from a Web client process wherein said request includes customer ID information;

spawning a program element operable on the computing node to process said request;

associating said customer ID information with the spawned program; and

allocating computing resources of said computing node to the spawned program element in accordance with said customer ID information associated with said request.

2. (Previously Presented) The method of claim 1 wherein the step of allocating comprises allocating a minimum level of resources to the spawned program element in accordance with said customer ID information.

3. (Previously Presented) The method of claim 1 wherein the step of allocating comprises allocating a maximum level of resources to the spawned program element in accordance with said customer ID information.

4. (Original) The method of claim I wherein said customer ID information is encoded in a process name of each said spawned program element.

5. (Original) The method of claim 1 wherein said computing resources includes processor time utilization.

6. (Original) The method of claim 1 wherein said computing resources includes main memory utilization.

7. (Original) The method of claim 1 wherein said computing resources includes secondary storage bandwidth utilization.

8. (Original) A system for delivery of services in a client/server distributed environment comprising: a server computing node; a server process operable on said server computing node for processing requests from a plurality of client processes coupled to said server computing node; a plurality of server child processes operable on said server computing node and spawned by said server process to process said requests from said plurality of client processes wherein each child process of said plurality of server child processes is associated with customer ID information; and a process resource manager operable on said server computing node to control allocation of resources of said server computing node among said plurality of server child processes wherein said resource manager is operable to control allocation of said resources in accordance with said customer ID information associated with said each child process.

9. (Original) The system of claim 8 wherein said server process is a Web server process, and wherein said each child process is a cgi-bin process.

10. (Previously Presented) The system of claim 8 wherein said process resource monitor further comprises:

a CPU time resource monitor element for allocating CPU time to said each child process;

a secondary storage bandwidth resource monitor for allocating secondary storage bandwidth to said each child process; and

a main memory resource monitor for allocating main memory to said each child process.

11. (Original) A system for providing Web services comprising:

means for receiving a request from a Web client process wherein said request includes customer ID information;

means for spawning a program element operable on a computing node to process said request;

means for associating said customer ID information with the spawned program; and

means for allocating computing resources of said computing node to the spawned program element in accordance with said customer ID information associated with said request.

12. (Original) The system of claim 11 wherein said means for allocating comprises:

means for allocating a minimum level of resources to the spawned program element in accordance with said customer ID information.

13. (Original) The system of claim 11 wherein said means for allocating comprises:

means for allocating a maximum level of resources to the spawned program element in accordance with said customer ID information.

14. (Original) The system of claim 11 wherein said customer ID information is encoded in a process name of each said spawned program element.

15. (Original) The system of claim 11 wherein said means for allocating includes means for allocating processor time utilization.

16. (Original) The system of claim 11 wherein said means for allocating includes means for allocating main memory utilization.

17. (Original) The system of claim 11 wherein said means for allocating includes means for allocating secondary storage bandwidth utilization.

IX. EVIDENCE APPENDIX

None.

X. RELATED PROCEEDINGS APPENDIX

None